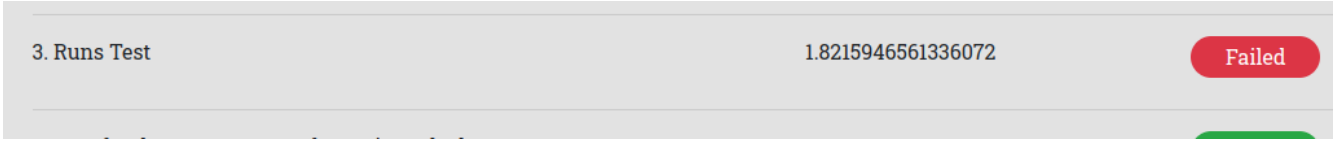
Sprawozdanie do zad. dom. nr 3

Zad. 1.:

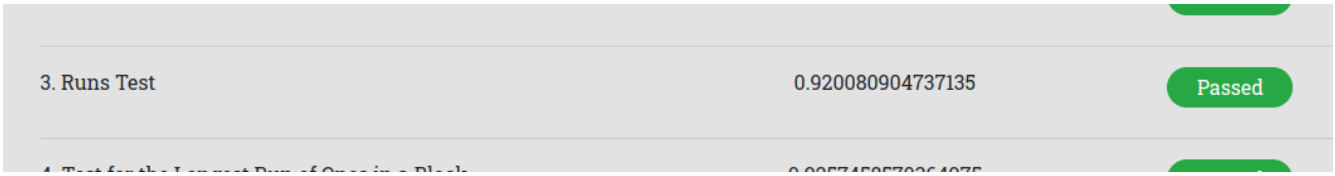
a) output generatora LCG, konkretnie z wykorzystaniem funkcji rand() w języku C:

ciągi 0-1 wygenerowane przy pomocy funkcji rand() nie przechodzą wszystkich testów NIST; w szczególności zazwyczaj niepoprawny wynik zauważyć można w teście nr 3 – "Runs Test".



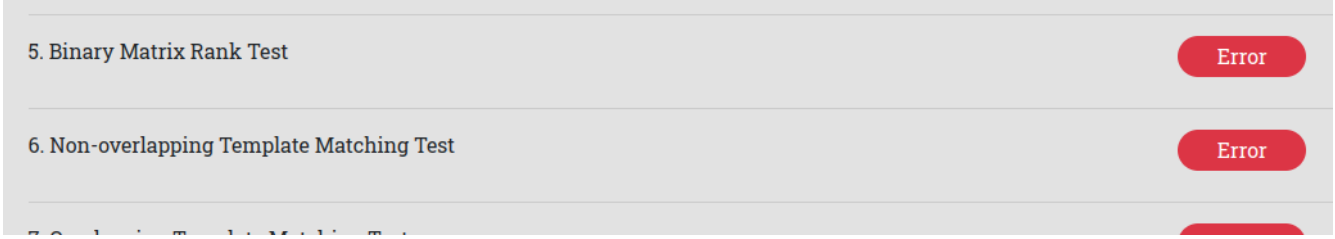
b) output generatora mt19937 w języku C++:

ciągi 0-1 wygenerowane przy pomocy generatora Mersenne Twister w większej części wykonanych prób przechodzą wszystkie testy NIST.



c) output SHA-1:

po zmianie wyniku SHA-1 na liczbę binarną, ponad połowa testów kończy się błędem z uwagi na zbyt krótkie wejście. Pozostałe testy kończą się sukcesem.

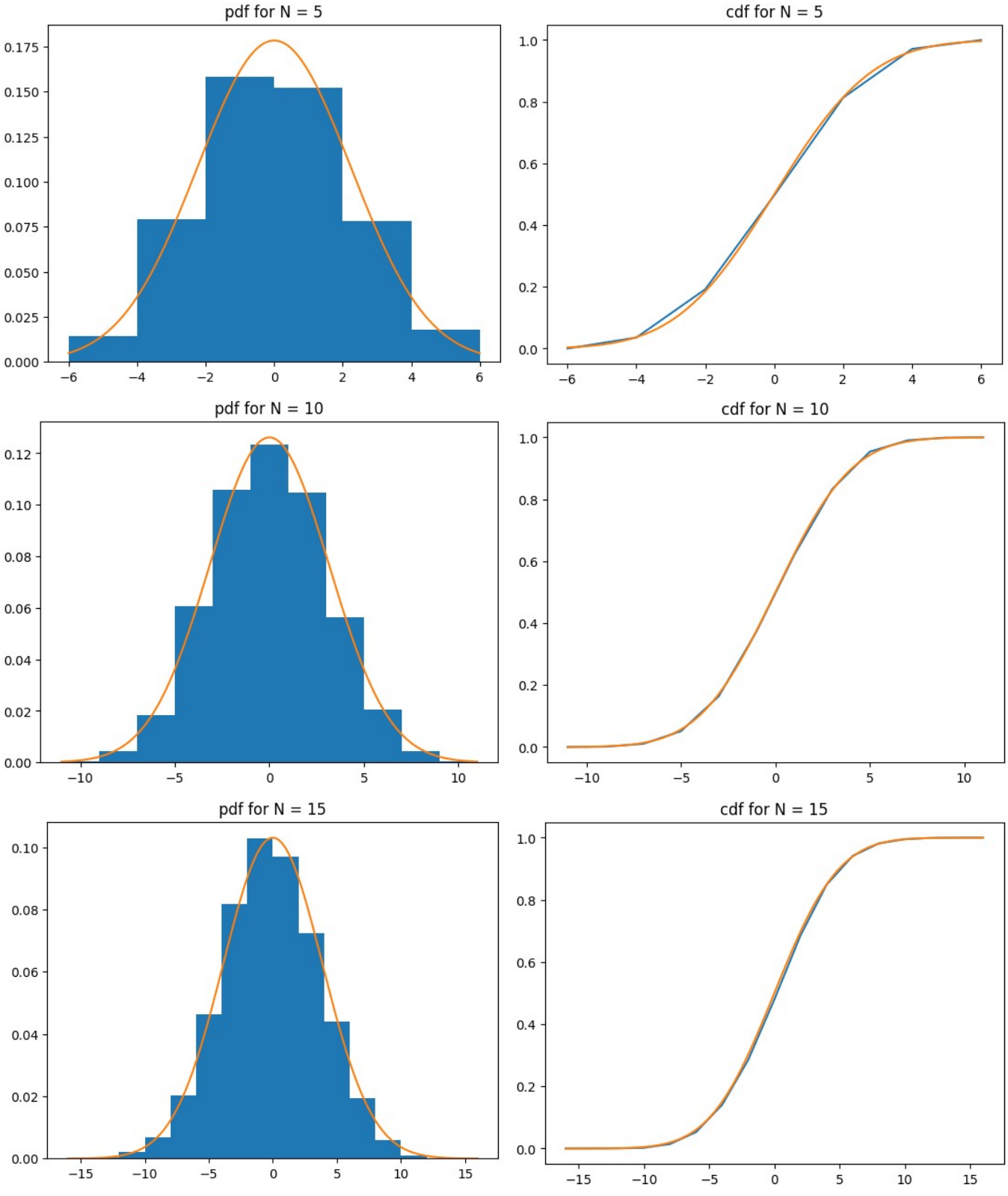


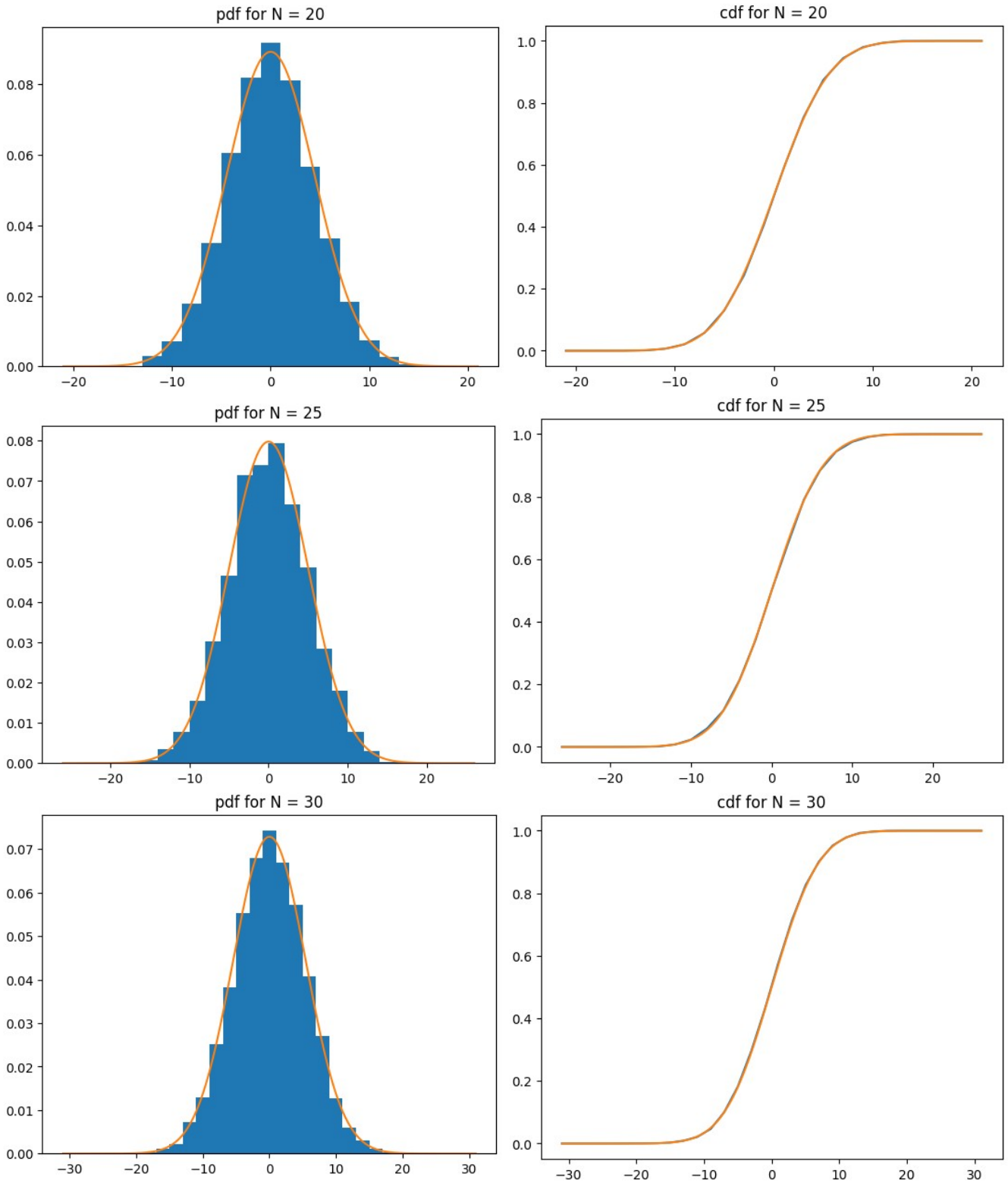
Wnioskujemy, że generator Mersenne Twister posiada pożądane właściwości generatora liczb pseudolosowych, przez co generuje bity z mniejszym obciążeniem niż generator LCG. Z tego powodu okazuje się lepszy do poważniejszych zastosowań, gdzie LCG może zawieść z powodu swojego obciążenia.

Podobnie, nie powinniśmy stosować funkcji haszującej do generowania liczb pseudolosowych, ponieważ haszowanie nie gwarantuje jednolitego rozkładu prawdopodobieństwa otrzymania konkretnego hasza jako wynik funkcji, przez co może charakteryzować się obciążeniem.

Zad. 2.:

a) Poniżej widoczne są wyniki numerycznych eksperymentów w postaci wykresów gęstości i dystrybunaty dla N równo kolejno 5, 10, 15, 20, 25, 30. niebieskie słupki/linie oznaczają eksperymentalnie uzyskane wyniki, natomiast na pomarańczowo zaznaczone są gęstość/dystrybuanta rozkładu normalnego z σ = sqrt(N):

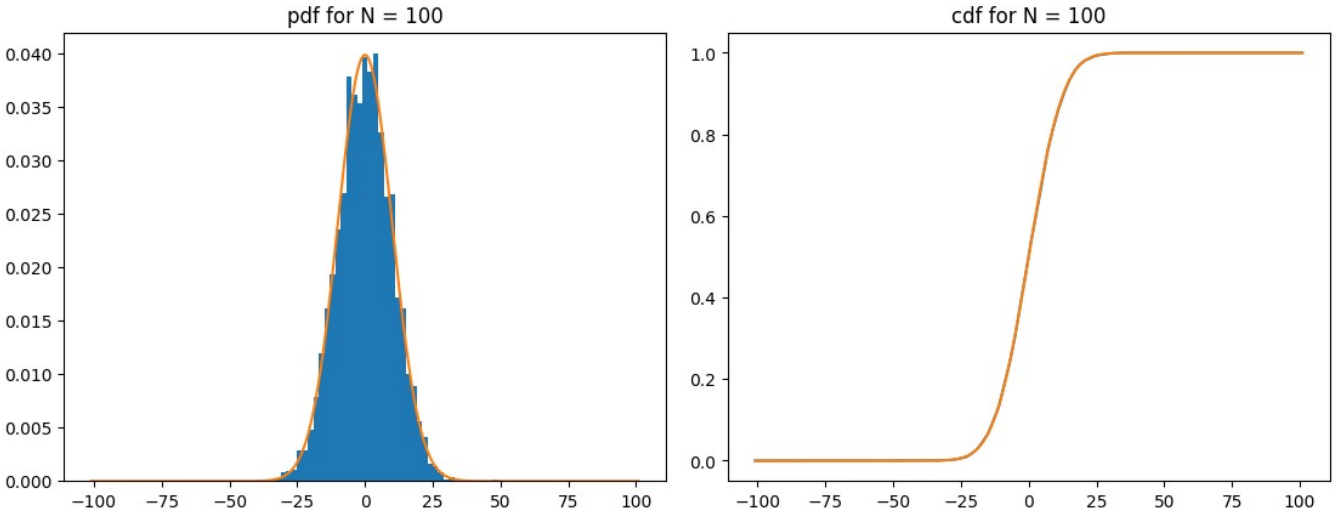




gdzie histogramy zostały znormalizowane, aby przybliżyć gęstość prawdopodobieństwa.

b) Na podstawie powyższych wyników możemy wywnioskować, że suma zmiennych losowych S_N przybliża rozkład normalny, przy czym dokładność estmacji rośnie wraz ze wzrostem N, zgodnie z Centralnym Tw. Granicznym.

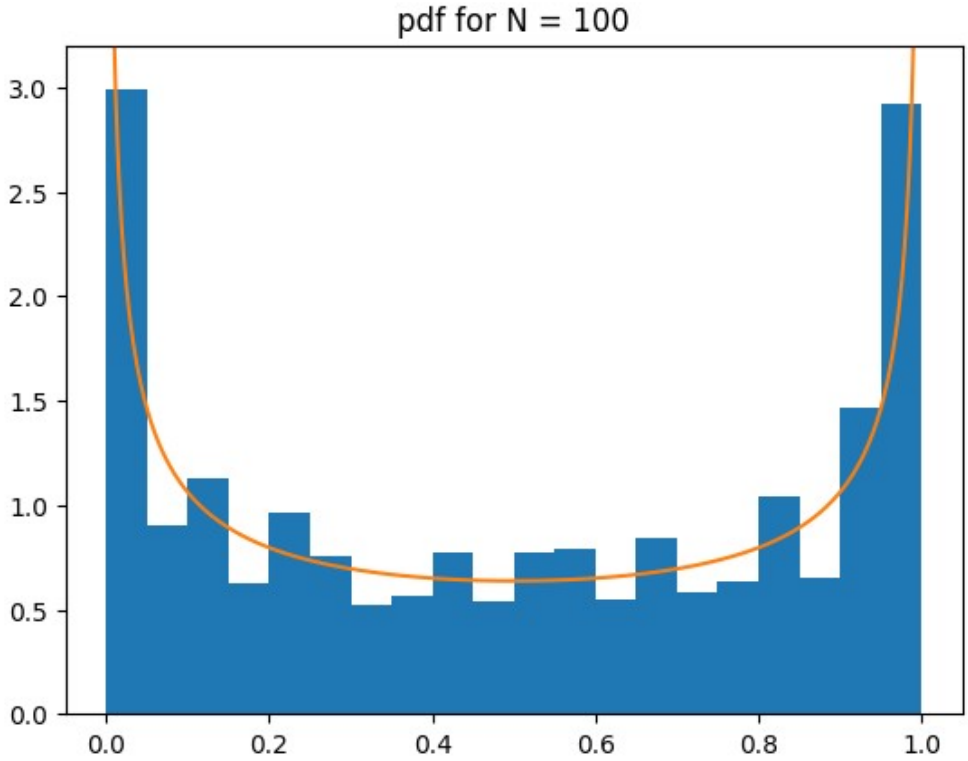
c) Ponawiając eksperyment dla N = 100, otrzymujemy poniższe wykresy:

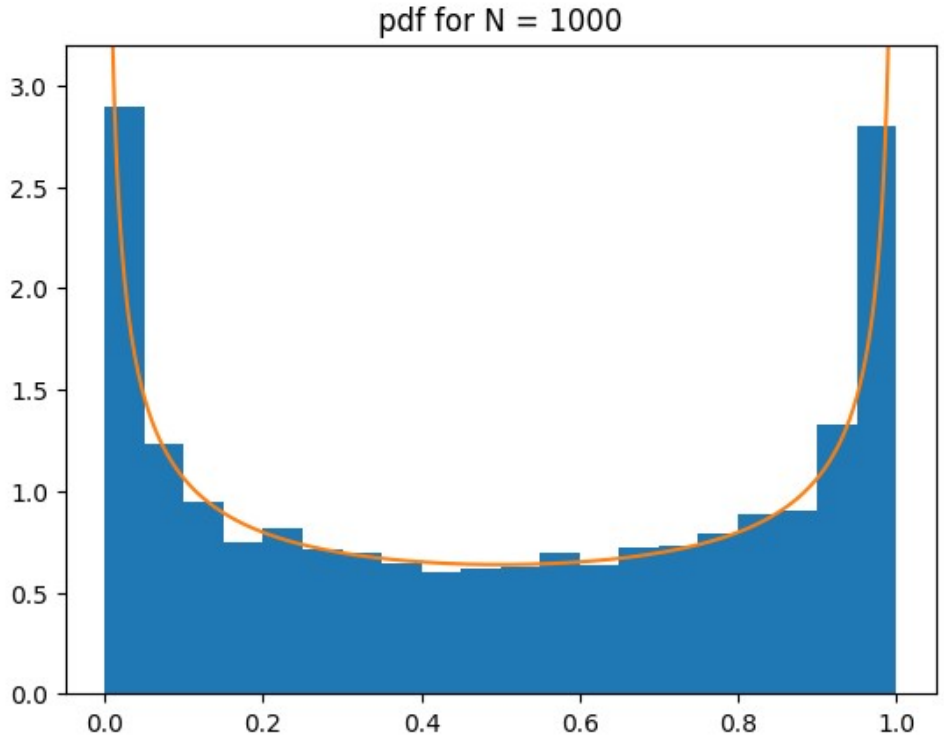


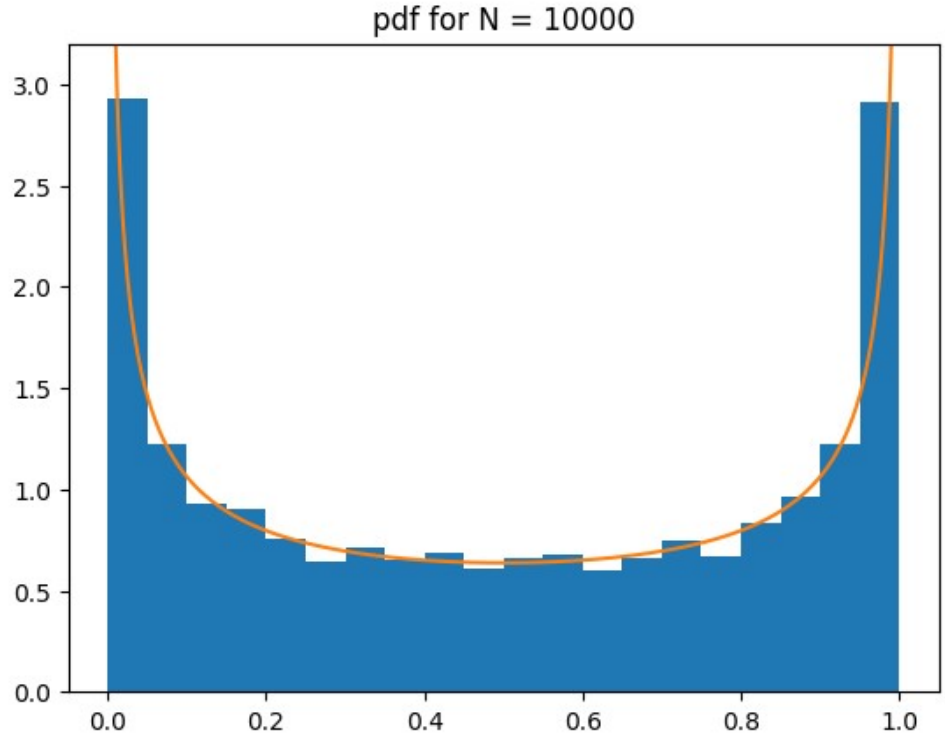
Zauważamy że, podobnie jak wcześniej, rozkład prawdopodobieństwa S_N przybliża rozkład normalny z wysoką dokładnością.

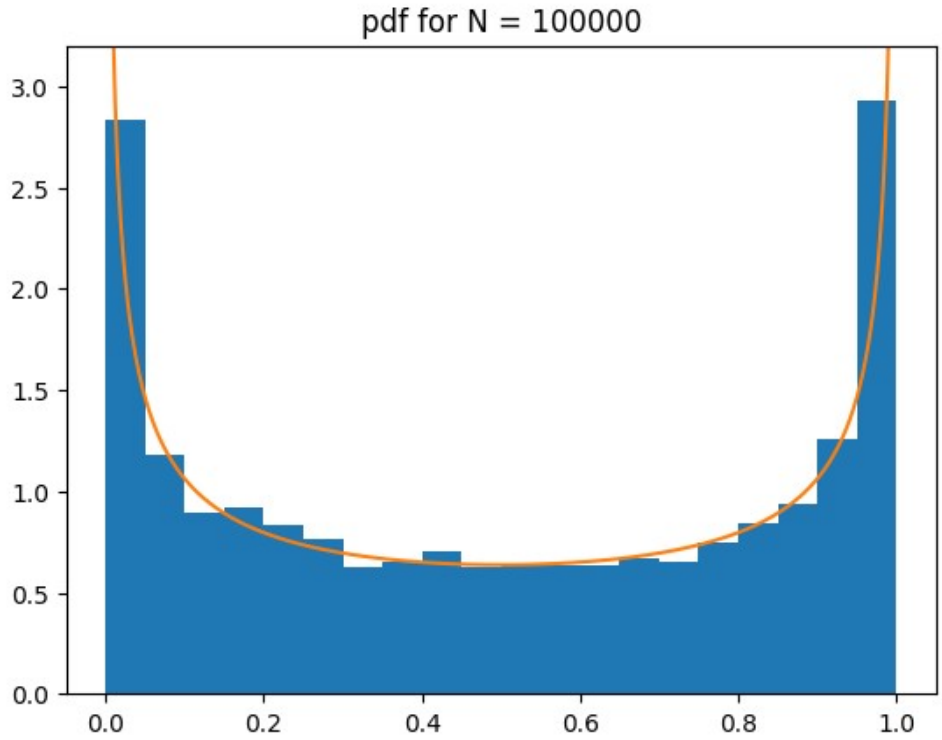
Zad. 3.:

Po przeprowadzeniu numerycznych eksperymentów dla k = 10000 oraz N równego kolejno 100, 1000, 10000, 100000, otrzymujemy dane z których możemy wygenerować poniższe histogramy. Niebieskie słupki oznaczają wyniki eksperymentów, natomiast na pomarańczowo zaznaczony został wykres gęstości rozkładu arcusa sinusa.




pdf for N = 1000
3.0
2.5
2.0
1.5
1.0
0.5
0.0
0.0
0.2
0.4
0.6
0.8
1.0


pdf for N = 10000
3.0
2.5
2.0
1.5
1.0
0.5
0.0
0.0
0.2
0.4
0.6
0.8
1.0



gdzie histogramy zostały znormalizowane, aby przybliżyć gęstość prawdopodobieństwa.

Na podstawie powyższych wykresów możemy wywnioskować, że uzyskany rozkład przybliża rozkład arcusa sinusa z wysoką precyzją. Oznacza to, że sytuacja, w której iloraz czasu spędzonego nad osią X do czasu spędzonego pod osią X jest bliski 1, posiada małe prawdopodobieństwo. Najbardziej prawdopodobne jest spędzenie znacznej większości czasu albo nad osią X, albo pod osią X. Jednym z płynących z tego wniosków jest to, że liczba przecięć z osią X jest mała.

Zauważamy także symetrię rozkładu względem prostej y = 0.5. Jest to intuicyjnie jasne, ponieważ możemy skonstruować oczywistą bijekcję między ścieżkami błądzenia losowego kończącymi się w punkcie (N, k) oraz (N, -k) poprzez odbicie lustrzane wokół osi X. Widać wobec tego, że prawdopodobieństwo spędzenia czasu t nad osią X wynosi tyle samo ile spędzenie czasu t pod osią X.

Warto również zauważyć, że z zad. 2. wiemy że najbardziej prawdopodobną sytuacją jest ta, w której błądzenie losowe skończy się blisko osi X, mimo że liczba przecięć z samą osią jest niewielka.